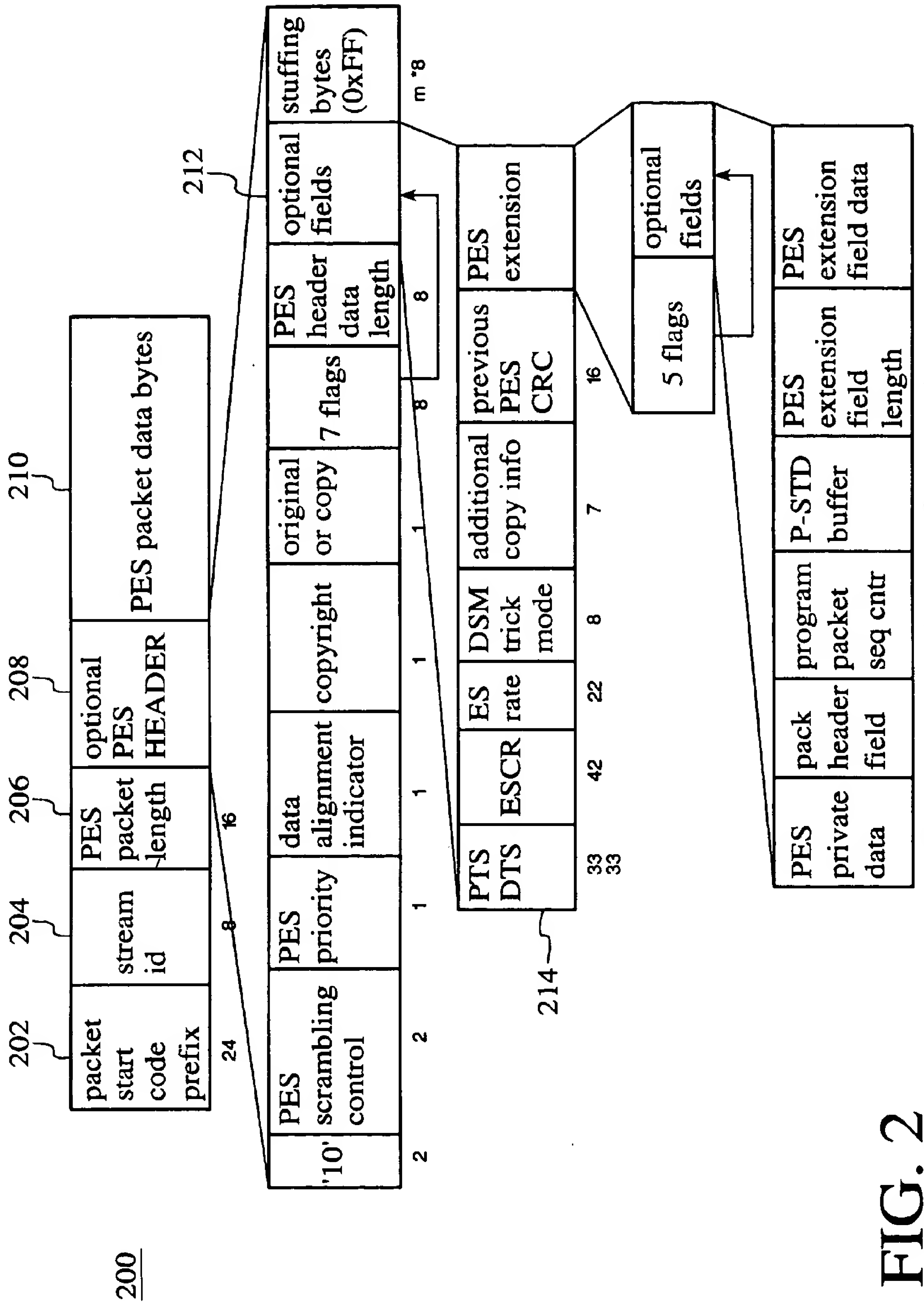
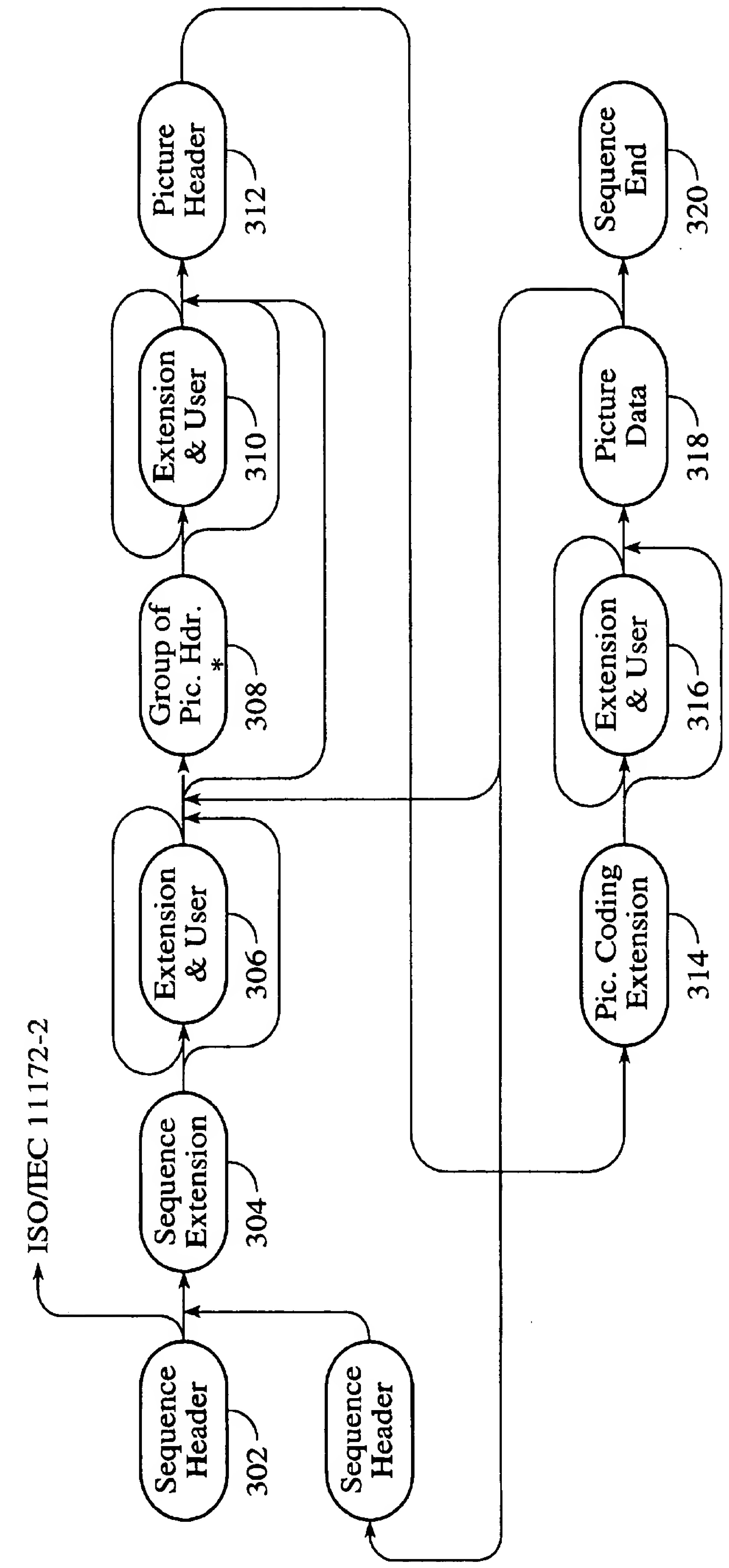


FIG. 2
(PRIOR ART)

EXHIBIT B



* After a GOP the first picture shall be an I-picture

FIG. 3A
(PRIOR ART)

video_sequence0 { 300	No. of bits	Mnemonic
next_start_code0		
sequence_header0 302		
if (nextbits0 = extension_start_code) {		
sequence_extension0 304		
do {		
extension_and_user_data(0) 306		
do {		
if (nextbits0 = group_start_code) {		
308 group_of_pictures_header0		
310 extension_and_user_data(1)		
}		
312 picture_header0		
314 picture_coding_extension0		
316 extensions_and_user_data(2)		
318 picture_data0		
} while ((nextbits0=picture_start_code) ǁ		
(nextbits0 = group_start_code))		
if (nextbits0! = sequence_end_code) {		
sequence_header0		
sequence_extension0		
}		
} while(nextbits0!=sequence_end_code)		
} else {		
/*ISO/IEC 11172-2 */		
}		
sequence_end_code 320	32	bslbf
}		

FIG. 3B
(PRIOR ART)

picture_header0 { 312	No. of bits	Mnemonic
picture_start_code 502	32	bslbf
temporal_reference 504	10	uimsbf
picture_coding_type	3	uimsbf
vbv_delay	16	uimsbf
if(picture_coding_type = 2 || picture_coding_type = 3) {		
full_pel_forward_vector	1	
forward_f_code	3	uimsbf
}		
if (picture_coding_type = 3) (		
full_pel_backward_vector	1	
backward_f_code	3	uimsbf
}		
while (nextbits0 = '1') {		
extra_bit_picture /* with the value '1' */	1	uimsbf
extra_information_picture	8	
}		
extra_bit_picture /* with the value '0' */	1	uimsbf
next_start_code0		
}		

FIG. 5 (PRIOR ART)

group_of_pictures_header0 { 308	No. of bits	Mnemonic
group_start_code 604	32	bslbf
time_code	25	bslbf
closed_gop 602	1	uimsbf
broken_link	1	uimsbf
next_start_code0		
}		

FIG. 6 (PRIOR ART)

At encoder input 1602	1 I	2 B	3 B	4 P	5 B	6 B	7 P	8 B	9 B	10 I	11 B	12 B	13 P
At encoder output 1604	1 I	4 P	2 B	3 B	7 P	5 B	6 B	10 I	8 B	9 B	13 P	11 B	12 B
At decoder output 1606	1 I	2 B	3 B	4 P	5 B	6 B	7 P	8 B	9 B	10 I	11 B	12 B	13 P

FIG. 16